Case C3.3: Taylor-Green vortex

Giorgio Giangaspero[*], Edwin van der Weide[†], Magnus Svärd[‡], Mark H. Carpenter[§] and Ken Mattsson[¶]

I. Discretization, iterative method and hardware

See the appendices.

II. Case summary

This test case is intended to test the capability of the code to capture turbulence accurately. The initial data is smooth and 3-dimensional and it will transition to turbulence. The initial state is given by

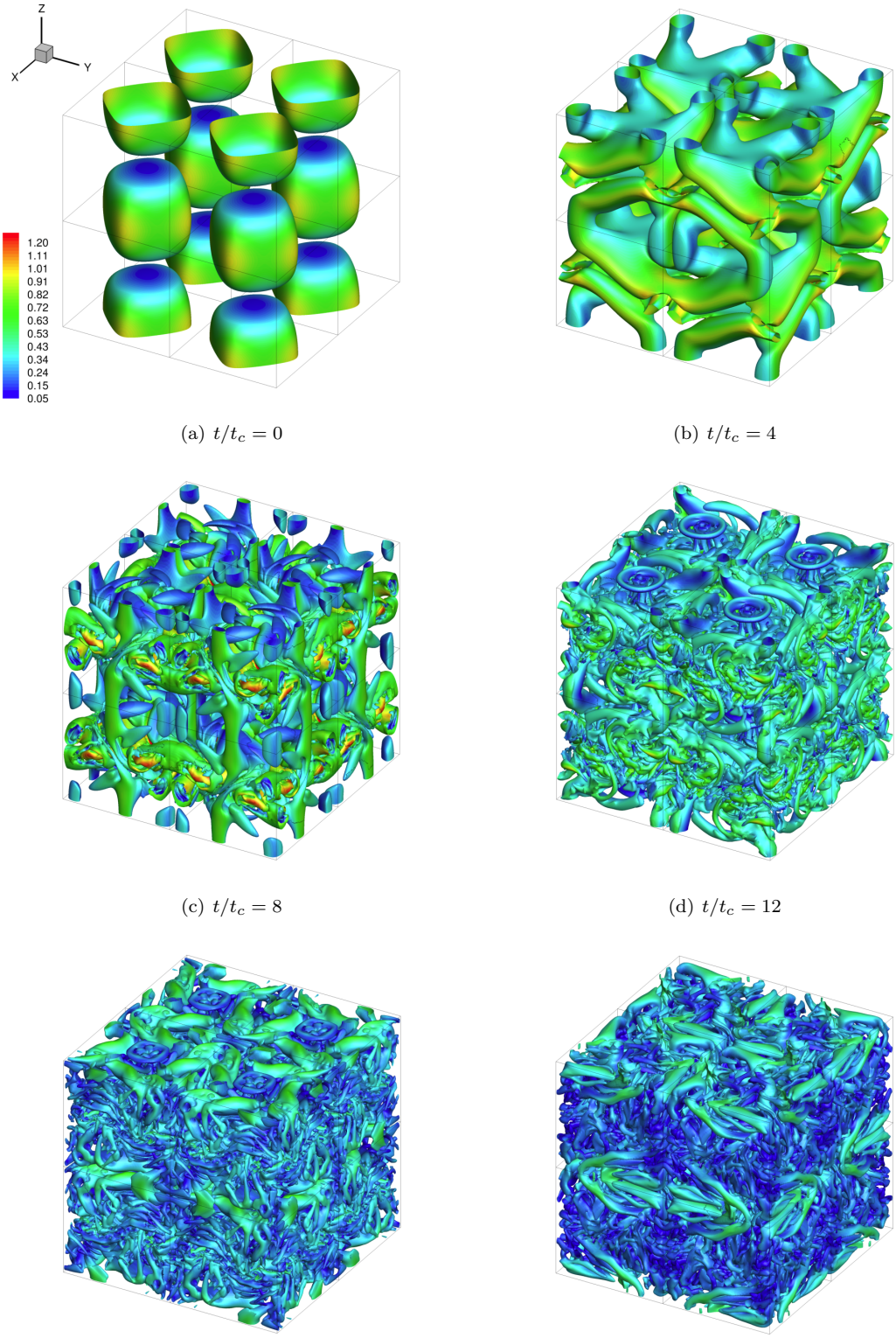
$$
\begin{aligned}
u &= V_0 \sin(\frac{x}{L}) \cos(\frac{y}{L}) \sin(\frac{z}{L}) \\
v &= -V_0 \cos(\frac{x}{L}) \sin(\frac{y}{L}) \sin(\frac{z}{L}) \\
w &= 0 \\
p &= p_0 + \frac{\rho_0 V_0^2}{16} \left(\cos(\frac{2x}{L}) + \cos(\frac{2y}{L}) \right) \left(\cos(\frac{2z}{L}) + 2 \right) \\
\rho &= \frac{p}{R T_0}
\end{aligned}
$$

The flow is governed by the Navier-Stokes equations with a Prandtl number of 0.71, specific heat ratio $\gamma = 1.4$ and the bulk viscosity is assumed to be zero. Furthermore, the Mach number $V_0/c_0 = 0.1$ and the Reynolds number $Re = \frac{\rho_0 V_0 L}{\mu} = 1600$. The initial temperature is uniform, $T_0 = \frac{p_0}{\rho_0 R}$. The solution is computed on the periodic domain $\Omega = \{-\pi L \leq x, y, z \leq \pi L\}$ which is discretized using four uniform structured grids containing 65^3, 129^3, 257^3 and 513^3 vertices respectively. For the 65^3, 129^3 and 257^3 grids it was possible to use our local Linux cluster, while the 513^3 grid was run on up to 512 processors of the LISA machine of SARA, the Dutch Supercomputer Center.

With a convective time scale $t_c = \frac{L}{V_0}$, the final time in the simulation is $t_{final} = 20 t_c$. The classical 4^{th} order Runge-Kutta scheme is used for the time-derivative of the governing equations. The spatial part is computed with the 2^{nd} and 5^{th} order scheme on all grids but the finest, where only the 5^{th} order solution is generated. Since the domain is periodic in all three directions, the internal discretization is used for the entire domain. Hence, the 5^{th} order scheme, which features an 8^{th} order discretization in the interior, should provide an 8^{th} order solution and from now on it will be referred to as such. The computational cost and the computed number of time steps for each of the runs is presented in table 1. Again, the computational cost is expressed in terms of work units, that is the CPU time scaled to one processor and divided by the cost of TauBench. The number of time steps is chosen based on stability considerations.

The temporal evolution of the flow field is shown in figure 1, where the iso-surfaces of the Q-criterion $Q = 0.1(V_0/L)^2$, colored with the velocity magnitude V/V_0, are plotted (see [1] for comparison). The initial large eddies gradually evolve towards smaller turbulent structures without reaching an isotropic state, symmetries are still present in the last solution. The peak of kinetic energy dissipation occurs at $t/t_c \approx 8$, where the maximum values of the velocity magnitude are observed. Then, a decay phase starts during which the dissipation takes place at lower and lower rates.

[*]Department of Mechanical Engineering, University of Twente, the Netherlands, e-mail: g.giangaspero@utwente.nl
[†]Department of Mechanical Engineering, University of Twente, the Netherlands, e-mail: e.t.a.vanderweide@utwente.nl
[‡]Department of Mathematics, University of Bergen, Norway, e-mail: Magnus.Svard@math.uib.no
[§]NASA Langley Research Center, Hamption, VA, e-mail: mark.h.carpenter@nasa.gov
[¶]Uppsala University, Uppsala, Sweden, e-mail: ken.mattsson@it.uu.se



(a) $t/t_c = 0$ (b) $t/t_c = 4$

(c) $t/t_c = 8$ (d) $t/t_c = 12$

(e) $t/t_c = 16$ (f) $t/t_c = 20$

Figure 1. Taylor-Green test case: evolution of the iso-surfaces of the Q-criterion ($Q = 0.1(V_0/L)^2$) colored with the non-dimensional velocity magnitude V/V_0. The solution has been computed with the 8^{th} order scheme on the 257^3 grid.

Table 1. Number of time steps and computational cost for all runs for test case 3.3.

grid	scheme	♯ time steps	work units
65^3	2^{nd} Order	20238	3.74e+03
	8^{th} Order	20238	4.32e+03
129^3	2^{nd} Order	20238	2.81e+04
	8^{th} Order	20238	3.70e+04
257^3	2^{nd} Order	23476	3.09e+05
	8^{th} Order	23476	3.39e+05
513^3	8^{th} Order	50000	9.55e+06

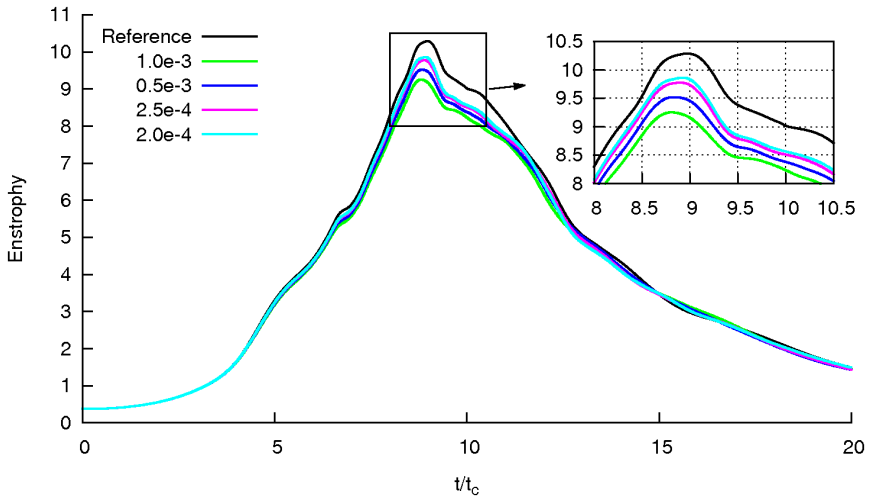


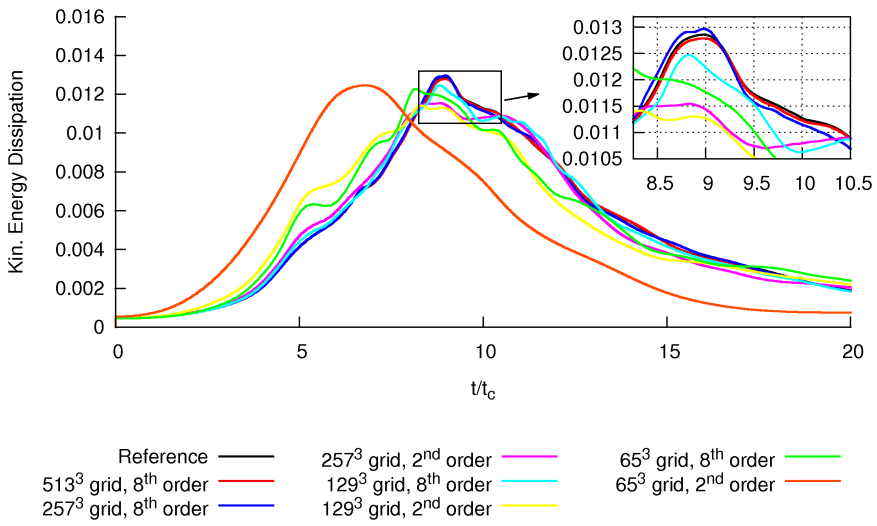
Figure 2. Enstrophy evolution with different artificial dissipation coefficients for test case C3.3 on the 257^3 grid, 8^{th} order scheme.

Our results are compared to a provided reference flow solution (see [2]). This reference solution was obtained with a dealiased pseudo-spectral code run on a 513^3 grid; time integration was performed with a low-storage 3-steps Runge-Kutta scheme and a non-dimensional time step of $1.0 \cdot 10^{-3}$. The solution consists in the temporal evolution of the following non-dimensional mean quantities:

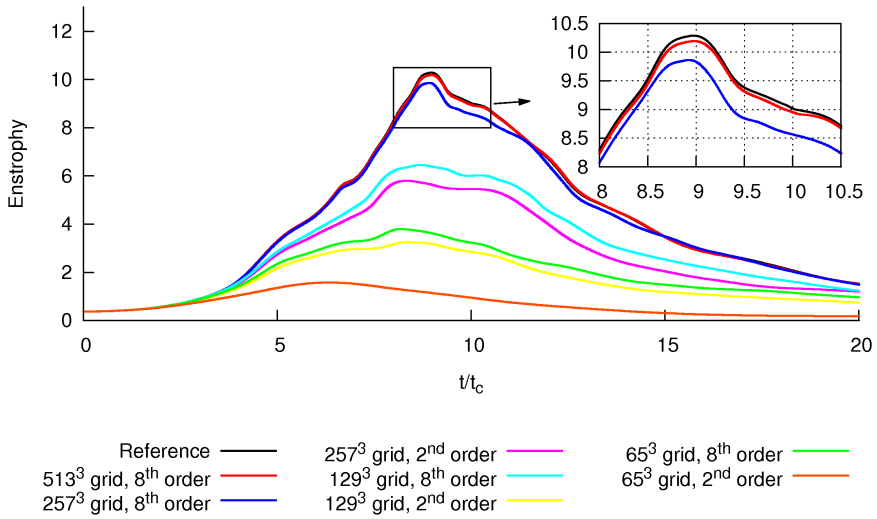
- the kinetic energy $E_k = \dfrac{1}{\rho_0 \Omega V_0^2} \displaystyle\int_\Omega \rho \frac{v \cdot v}{2} \mathrm{d}\Omega$;
- the dissipation of kinetic energy $\epsilon = -t_c \dfrac{\mathrm{d}E_k}{\mathrm{d}t}$;
- the enstrophy $\mathcal{E} = \dfrac{t_c^2}{\rho_0 \Omega} \displaystyle\int_\Omega \rho \frac{w \cdot w}{2} \mathrm{d}\Omega$.

A study of the influence of the artificial dissipation, see section A, was performed on the 256^3 grid using the 8^{th} order scheme. Figure 2 shows the enstrophy evolution obtained with different artificial dissipation coefficients down to the stability limit. It is clear that reducing the amount of artificial dissipation leads to a better solution.

The evolution of dissipation of kinetic energy and enstrophy are presented in figure 3 for the different schemes and grids used. Our solution on the fine grid is extremely close to the reference solution, which is also confirmed by the iso-contours of the vorticity in the plane $x/L = -\pi$ at $t/t_c = 8$, see figure 4. From figure 3 and from table 1 it is clear that the 8^{th} order scheme outperforms the 2^{nd} order scheme. For instance, the 8^{th} order solution on the 129^3 grid is more accurate (closer to the reference) than the 2^{nd} order solution on the 257^3 grid; at the same time, the computational cost of the former is almost one order of magnitude



(a) Kinetic energy dissipation



(b) Enstrophy

Figure 3. Evolution of the kinetic energy dissipation and the enstrophy for the different schemes and grids used for test case 3.3.

less then the latter (3.70e+04 and 3.09e+05 work units respectively). The same applies to the other grids. Therefore the 8^{th} order scheme gives a more accurate solution at a lower computational cost than the 2^{nd} order scheme, although the difference between the two is smaller than expected from a truncation error analysis (not reported here). However, design order refers to the asymptotic slope of the error of a smooth solution that is well resolved. The present DNS solutions are far from being resolved and at this resolution they can be viewed as non-smooth which explains the sub-optimal convergence rates. Nevertheless, the error is still smaller for the high-order scheme due to its superior ability to resolve sharp gradients.

A. Background information for the SBP-SAT scheme

As is well-known, stability of a numerical scheme is a key property for a robust and accurate numerical solution. Proving stability for high-order finite-difference schemes on bounded domains is a highly non-trivial task. One successful way to obtain stability proofs is to employ so-called Summation-by-Parts (SBP) schemes with Simultaneous Approximation Terms (SAT) for imposing boundary conditions. With a simple example, we will briefly describe how stability proofs can be obtained.

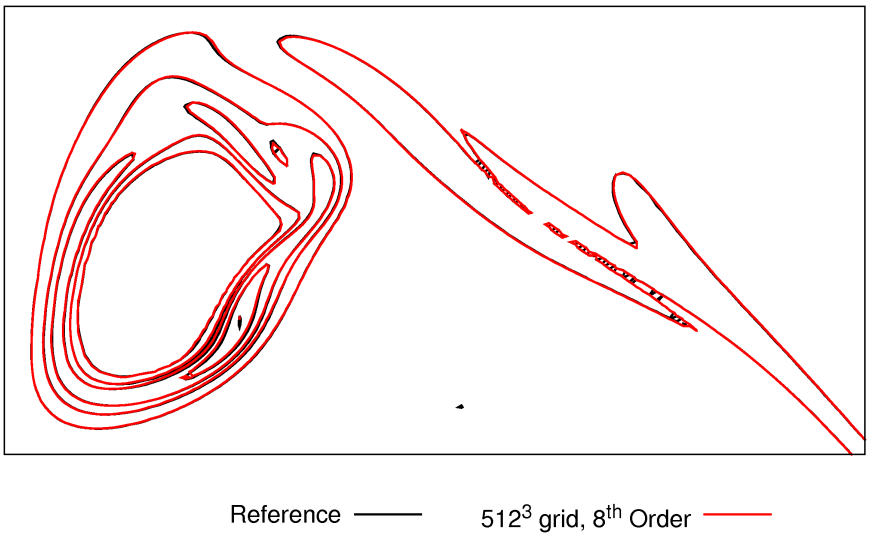


Figure 4. Comparison of the iso-contours of the dimensionless vorticity norm on the periodic face $x/L = -\pi$ at $t/t_c = 8$.

Consider the scalar advection equation,

$$\begin{aligned} &u_t + au_x = 0, \quad 0 < x < 1, \quad 0 < t \leq T \\ &a^+ u(0,t) = a^+ g_l(t) \\ &a^- u(1,t) = a^- g_r(t) \end{aligned} \tag{1}$$

where $a^+ = \max(a, 0)$ and $a^- = \min(a, 0)$. Furthermore, we augment the equation with initial data $u(x, 0) = f(x)$, bounded in L^2. To demonstrate well-posedness, we employ the energy method.

$$\begin{aligned} &\|u\|_t^2 + a \int_0^1 uu_x \, dx = 0 \\ &\|u\|_t^2 \leq au^2(0,t) - au^2(1,t) \leq a^+ g_l(t)^2 - a^- g_r(t)^2 \end{aligned} \tag{2}$$

Integrating in time gives the bound

$$\|u(\cdot, T)\| \leq \|f\| + a^+ \int_0^T g_l(t)^2 \, dt - a^- \int_0^T g_r(t)^2 \, dt. \tag{3}$$

For linear PDEs, such a bound is sufficient to prove well-posedness.

Next, we turn to the SBP-SAT semi-discretization of (1). To this end, we introduce the computational grid, $x_i = ih$, $i \in \{0, 1, 2, ..., N\}$ and $h > 0$ is the grid spacing. For the moment, we keep time continuous. With each grid point x_i, we associate a value $v_i(t)$, and define a grid function $v(t) = (v_0, v_1, v_2, ...)^T$. The SBP difference operator, D is a matrix with the following properties: $D = P^{-1}Q$ where P and Q are two matrices; $P = P^T > 0$ and $Q + Q^T = B = \text{diag}(-1, 0, ..., 0, 1)$. The matrix P can be used to define a weighted l^2 equivalent norm as $\|v\|^2 = v^T P v$. We will also need the vectors $e_0 = (1, 0, 0..., 0)^T$ and $e_N = (0, ..., 0, 1)^T$.

Let w denote a smooth function and define a grid function $\bar{w} = (w(x_0), ..., w(x_N))^T$ and $\bar{w}_x = (w_x(x_0), ..., w(x_N))^T$. It turns out that the SBP property precludes the accuracy of D to be uniform in space. We have

$$D\bar{w} = \bar{w}_x + \bar{T}$$

where $\bar{T}$ is the truncation error. In general, it takes the form,

$$\bar{T}^T = (\mathcal{O}(h^s), ..., \mathcal{O}(h^s), \mathcal{O}(h^p), ..., \mathcal{O}(h^p), \mathcal{O}(h^s), ..., \mathcal{O}(h^s)). \tag{4}$$

where $s < p$ and the lower accuracy is confined to a few (finite) number of points close to the boundary. SBP operators exist with various orders of accuracy, [3]. In particular, if P is a diagonal matrix, there are

SBP operators with p even and $p \leq 8$, and $s = p/2$. If P is allowed to have off-diagonal elements for a few points near the boundary $s = p - 1$ can be achieved.

Using the SBP operators, we now define a semi-discrete scheme for (1).

$$v_t + aDv = \sigma_l a^+ P^{-1} e_0 (v_0 - g_l(t)) + a^- \sigma_r P^{-1} e_N (v_N - g_r(t))$$

The right-hand side are the SAT:s, which impose the boundary conditions weakly. (Originally proposed in [4].) $\sigma_{l,r}$ are two scalar parameters, to be determined by the stability analysis. Multiplying by $2v^T P$, we obtain

$$\|v\|_t^2 - a(v_0^2 - v_N^2) = 2\sigma_l a^+ v_0 (v_0 - g_l(t)) + 2a^- \sigma_r v_N (v_N - g_r(t)) \tag{5}$$

For stability, it is sufficient to obtain a bound with $g_{l,r} = 0$. In that case, it is easy to see that we must require $\sigma_l \leq -1/2$ and $\sigma_r \geq 1/2$ to obtain a bounded growth of $\|v\|$. More generally, allowing boundary data to be inhomogeneous when deriving a bound leads to *strong stability*. (See [5]. The benefit of proving strong stability as opposed to stability is that less regularity in the boundary data is required.) For strong stability, it can be shown that $\sigma_{l,r}$ must satisfy $\sigma_l < -1/2$ and $\sigma_r < 1/2$, i.e., strict inequalities. As an example, the choice $\sigma_l = -1, \sigma_r = 1$ leads to

$$\|v\|_t^2 - a(v_0^2 - v_N^2) = -2a^+ v_0 (v_0 - g_l(t)) + 2a^- v_N (v_N - g_r(t))$$

or

$$\|v\|_t^2 \leq -a^+ (v_0 - g)^2 + a^+ g_l(t)^2 + a^- (v_N - g_r(t)^2) - a^- g^2 \tag{6}$$

If $v_0 = g_l, v_N = g_r$, (6) is the same as (2), but this is not the case and the additional terms add a small damping to the boundary. Upon integration of (6) in time, an estimate corresponding to (3) is obtained. We also remark that the SAT terms are accurate as they do not contribute to a truncation error in the scheme. Furthermore, semi-discrete stability guarantees stability of the fully discrete problem obtained by employing Runge-Kutta schemes in time, [6].

The above example, demonstrates the general procedure for obtaining energy estimates for an SBP-SAT scheme. Naturally, for systems of PDEs, in 3-D with stretched and curvilinear multi-block grids, and with additional parabolic terms, the algebra for proving stability becomes more involved. However, the resulting schemes are still fairly straightforward to use. For the linearized Euler and Navier-Stokes equations, semi-discrete energy estimates have been derived. (See [7–9] and references therein.) Different boundary types, including far-field, walls and grid block interfaces are included in the theory. For flows with smooth solutions, linear stability implies convergence as the grid size vanishes. (See [10].)

B. Code description

Both a general code and specialized codes for some of the test cases (used in the 1st and 2nd high order workshop, see [11]) are available. The general code is a 3D code that can handle multiblock grids and can run on (massively) parallel platforms. For load balancing reasons the blocks are split during runtime in an arbitrary number of sub-blocks with a halo treatment of the newly created interfaces, such that the results are identical to the sequential algorithm.

The specialized codes assume a single block 2D grid and do not have parallel capabilities, hence they are relatively easy to modify for testing purposes. Due to the fact that these codes can only be used for one specific test case and the fact that the general purpose code can only handle 3D problems, the efficiency of the specialized codes is quite a bit higher than the general purpose code.

The discretization schemes used are finite difference SBP-SAT schemes, see section A, of order 2 to 5. Thanks to the energy stability property of these schemes no or a significantly reduced amount of artificial dissipation is needed compared to schemes which do not posses this (or a similar) property. This leads to a higher accuracy of the numerical solutions.

For the steady test cases the set of nonlinear algebraic equations is solved using the nonlinear solver library of PETSc [12]. This library requires the Jacobian matrix of the spatial residual, which is computed via dual numbers [13] and appropriate coloring of the vertices of the grid, for which the PETSc routines are

used. Initial guesses are obtained via grid sequencing, where appropriate. The solution of the linear systems needed by PETSc's nonlinear solution algorithm is obtained by Block ILU preconditioned GMRES.

Implicit time integration schemes of the ESDIRK type [14] are available, for which the resulting nonlinear systems are solved using a slightly adapted version of the steady state algorithm explained above. However, for the unsteady test cases considered, the Euler vortex and the Taylor-Green vortex, the time steps needed for accuracy are relatively small compared to the stability limit of explicit time integration schemes and therefore the explicit schemes are better suited for these cases. The available explicit schemes are the classical 4^{th} order Runge Kutta scheme (RK4, [15]) and TVD Runge Kutta schemes up till 3^{rd} order [16]. As the maximum CFL number of the RK4 scheme is significantly higher than the CFL number of the TVD Runge Kutta schemes, the RK4 scheme is used for the unsteady test cases mentioned above.

For the post processing standard commercially available software, such as Tecplot, and open-source software, such as Gnuplot, are used. Grid adaption has not been carried out.

C. Machines description

The results for the easy test cases have been obtained on a Linux work station running Ubuntu 10.04 with an Intel i7-2600 CPU running at 3.4 GHz, with 8 Mb of cache. The machine contains 16 Gb of RAM memory with an equivalent amount of swap. Running the Taubench on this machine led to a CPU time of 5.59 seconds (average over 4 runs).

The difficult test cases were run on up to 512 processors on the LISA machine of SARA, the Dutch Supercomputing Center and Hexagon, the Cray XE6 machine of the University of Bergen. Running the Taubench on these machines led to a CPU time of 10.3 and 10.8 seconds respectively (average over 4 runs).

References

[1] Morris, C., "Studies of Inviscid Flux Schemes for Acoustics and Turbulence Problems," *AIAA paper 2013-0075*, 2013.

[2] "2nd International Workshop on High-Order CFD Methods, Cologne, Germany," May 2013, http://www.dlr.de/as/hiocfd.

[3] Strand, B., "Summation by Parts for Finite Difference Approximations for d/d x," *J. Comput. Phys.*, Vol. 110, 1994.

[4] Carpenter, M. H., Gottlieb, D., and Abarbanel, S., "Time-stable boundary conditions for finite-difference schemes solving hyperbolic systems: Methodology and application to high-order compact schemes," *J. Comput. Phys.*, Vol. 111(2), 1994.

[5] Gustafsson, B., Kreiss, H.-O., and Oliger, J., *Time dependent problems and difference methods*, John Wiley & Sons, Inc., 1995.

[6] Kreiss, H.-O. and Wu, L., "On the stability definition of difference approximations for the initial boundary value problem," *Applied Numerical Mathematics*, Vol. 12, 1993, pp. 213–227.

[7] Svärd, M., Carpenter, M., and Nordström, J., "A stable high-order finite difference scheme for the compressible Navier-Stokes equations, far-field boundary conditions," *Journal of Computational Physics*, Vol. 225, 2007, pp. 1020–1038.

[8] Svärd, M. and Nordström, J., "A stable high-order finite difference scheme for the compressible Navier-Stokes equations, no-slip wall boundary conditions," *J. Comput. Phys.*, Vol. 227, 2008, pp. 4805–4824.

[9] Nordström, J., Gong, J., van der Weide, E., and Svärd, M., "A stable and conservative high order multi-block method for the compressible Navier-Stokes equations," *J. Comput. Phys.*, Vol. 228, 2009, pp. 9020–9035.

[10] Strang, G., "Accurate partial difference methods II. Non-linear problems," *Num. Math.*, Vol. 6, 1964, pp. 37–46.

[11] van der Weide, E., Giangaspero, G., and Svärd, M., "Efficiency Benchmarking of an Energy Stable High-Order Finite Difference Discretization," *AIAA Journal*, 2015, doi: 10.2514/1.J053500.

[12] Balay, S., Brown, J., Buschelman, K., Eijkhout, V., Gropp, W., Kaushik, D., Knepley, M., McInnes, L. C., Smith, B., and H.Zhang, "PETSc Users Manual, Revision 3.2," Tech. rep., Argonne National Laboratory, 2011.

[13] Fike, J. A., Jongsma, S., Alonso, J., and v.d. Weide, E., "Optimization with Gradient and Hessian Information Calculated Using Hyper-Dual Numbers," *AIAA paper 2011-3807*, 2011.

[14] Kennedy, C. and Carpenter, M. H., "Additive Runge-Kutta schemes for convection-diffusion-reactio equations," *Applied Numerical Mathematics*, Vol. 44, 2003, pp. 139–181.

[15] Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P., *Numerical Recipes: The Art of Scientific Computing*, Cambridge University Press, 3rd ed., 2007.

[16] Gottlieb, S., Shu, C. W., and Tadmor, E., "High order time discretizations with strong stability property," *SIAM Review*, Vol. 43, 2001, pp. 89 – 112.